Identifying parameters of local switching models: a geometrical approach

David W. Pearson* and José Ragot**
* EURISE, Jean Monnet University of Saint-Etienne, 23 rue du Docteur Michelon, 42023 Saint-Etienne, France. Email david.pearson@univ-st-etienne.fr
** CRAN-INPL, 2, avenue de la Forêt de Haye, 54516 Vandœuvre, France. Email Jose.Ragot@ensem.inpl-nancy.fr

***Abstract*— In this paper we look at the problem where we have two mixed sets of data on the same scatter plot. We want to calculate linear regression lines for each of the two sets, but first of all we need to decide which data belong to which set. We propose one possible solution to this problem involving data classification and parameters estimation.**

Keywords: scatter plots, switching regression, mixture of models, change of regime, differential geometry.

I. Introduction

During the last few decades a number of methodological papers on models with discrete parameter shifts have been published and revised interest in the so-called regime switching model. Since the pioneer work of Quandt (1958) [11], various attempts have been made to fit data to models with changes of regimes. The basic idea of regime switching models is that the process under consideration is time-invariant conditional on a regime variable which indicates the regime prevailing at each time. This kind of representation has been successfully developed in several scientific communities and mainly for multi-models [9], hybrid systems [3], time-series [8] and mixture models [5], [10].

There is a long history of work on change detection in time-series [2] and many studies assume that within each time-segment, the distribution of data does not depend on time. The most widely used assumption is the piecewise linear model in which each segment is described by a linear regression [4], [8]. In the field of data analysis, the same objective is underlying but the time is not a variable that can be used to describe the models. In this area, we find the well known problem of classification for which it is desired to separate the data into clusters and to describe each cluster by a model [5].

Our paper deals with the second class of problem and we are mainly interested with the problem of mixing regression equations. Our objective is only concerned with the estimation of the local model, the generating switching process itself is not here estimated ; however, as it will be seen later, we need to estimate the switching or the partitioning of the data. As is evident, if the partitioning is known, the problem of identifying the local models can easily be solved using standard techniques. However, when this partitioning is unknown the problem becomes much more difficult [12] unless the local models are known.

Our approach deals with a combined estimation allowing the partitioning or the data allocation to clusters and the local model parameters. In the paper, we consider a simpler version of the problem, where the number of local linear models and their order are a priori known. The contribution of the paper is to propose an original approach to solve a classical problem.

The basic problem that we are looking at then is fairly easy to state. Quite simply we assume that we are given a scatter plot, where there are obviously two fairly distinct data sets. We want to determine which data belong to which set before carrying out a regression analysis.

The solution presented in the following section is based on ideas from differential geometry and could be thought of as an information geometry approach to the problem [1]. Following the section 2 where our method is presented, the section 3 is devoted to numerical aspects of the calculations. Then, in section 4 some examples are presented before looking at conclusions and perspectives in section 5. There is also a technical annex to the paper where some of the tools from differential geometry are resumed.

II. A Possible Solution

A more precise statement of the problem is that we are given two data sets $S_1 = (x_i^1, x_i^2), i \in \mathcal{I}_1$ and $S_2 = (x_j^1, x_j^2), j \in \mathcal{I}_\in$ where $\mathcal{I}_1 = \{i_1 \dots i_{m_1}\}$ and

$\mathcal{I}_2 = \{j_1 \dots j_{m_2}\}$ are two subsets of index. The data are mixed together of course and we don't know the values of m_1 and m_2. All we know about the data is that they belong to two distinct sets and within each set they are related by a linear equation:

$$x^2 = \gamma x^1 + \delta \quad x \in S_1,\ x^2 = \lambda x^1 + \mu \quad x \in S_2 \quad (1a)$$

where γ and δ are unknown but different for each of the two sets. For numerical reasons we assume that the data are normalised and all lie within the square $\|x_i\| \leq \epsilon$ for some parameter $\epsilon > 0$.

Our idea is very simple, we map the points from the 2-dimensional space where the data lie into a higher dimensional space, in our case a 4-dimensional space (because we consider two data sets). We then carry out operations in the higher dimensional space that are designed to give indications as to whether or not two data points belong to the same set. There are, of course, various choices of mappings from lower to higher dimensional spaces. We choose a particular type of mapping coming from differential geometry and one that is particularly useful in nonlinear control theory [6]. First of all we need some notation. Let $a_k(v)$ for $k = 1, \dots, 4$ denote a vector field at the point $v \in \mathbb{R}^4$ and let $\alpha_k(t)v_0$ denote the flow of the vector field passing through the point v_0. In other words, this corresponds to the solution of the differential system

$$\dot{v}(t) = a_k(v(t)),\ v(0) = v_0 \quad (2)$$

where as usual $\dot{v} := \frac{dv}{dt}$. Let $\alpha_k(t)_\star$ denote the differential of the mapping $\alpha_k(t)$ (the jacobian matrix once a coordinate system is chosen). Two, or more, flows can be concatenated. For example $\alpha_j(t)\alpha_i(s)v_0$ means that we start at v_0 and go in the direction of a_i for s time units and then from the point $\alpha_i(s)v_0$ we go in the direction of α_j for t time units. This concatenation is symbolic, of course, because the vector fields may be nonlinear.

With these tools, we can now define a mapping as follows. Take any two points x_i, x_j of the whole set $\mathcal{I}_1 \cup \mathcal{I}_2$ and let z be the vector:

$$z = \begin{bmatrix} x_i^2 & x_i^1 & x_j^2 & x_j^1 \end{bmatrix}^T \quad (3)$$

Then, for a set of vector fields $\{a_1, \dots, a_4\}$, via their flows we define $\phi : \mathbb{R}^4 \to \mathbb{R}^4$ as follows:

$$\phi(z) = \alpha_4(z^4) \cdots \alpha_1(z^1)v_0 \quad (4)$$

In (2), the vector fields and initial point v_0 are parameters that we are going to calculate later on. It is known that for $\|z\|$ sufficiently small, ϕ is a diffeomorphism.

Let $\partial_i := \frac{\partial}{\partial z^i}$, then $\partial_i \phi(z)$ is simply the i^{th} column of the differential of ϕ evaluated at the point $\phi(z)$. Now, by the chain rule of differentiation [6], we have:

$$\partial_i \phi(z) = \alpha_4(z^4)_\star \cdots \alpha_{i-1}(z^{i-1})_\star \\ a_i(\alpha_i(z^i) \cdots \alpha_1(z^1)v_0) \quad (5)$$

The basis of our approach is to compare tangent vectors, but (5) gives us a tangent vector at the point $\phi(z)$ and we would like to compare them at the origin (the point $z = 0$). This is where the differentials $\alpha_k(-t)_\star$ come in useful, because this operator pulls a tangent vector at $\alpha_k(t)v_0$ (i.e. $a_k(\alpha_k(t)v_0)$ back to the origin $t = 0$ and we know that $\alpha_k(-t)_\star \alpha_k(t)_\star = 1$ where 1 indicates the identity matrix [6]. Let

$$\alpha(-z)_\star := \alpha_1(-z^1)_\star \cdots \alpha_4(-z^4)_\star \quad (6)$$

Then applying operator $\alpha(-z)_\star$ on $\partial_i \phi(z)$ will pull tangent vectors back to the origin. By repeated application of the CBH formula we can develop a multinomial approximation for any of the columns of the jacobian matrix of ϕ when pulled back to the origin. In our case, the most complicated expression is for $\partial_4 \phi$ which is:

$$\alpha(-z)_\star \partial_4 \phi = \sum_{k_1=0}^{\infty} \frac{(z^1)^{k_1}}{k_1!} ad_{a_1}^{k_1} \Big(\sum_{k_2=0}^{\infty} \frac{(z^2)^{k_2}}{k_2!} \\ ad_{a_2}^{k_2} \Big(\sum_{k_3=0}^{\infty} \frac{(z^3)^{k_3}}{k_3!} ad_{a_3}^{k_3}(a_4(v_0)) \Big) \Big) \quad (7)$$

Here, we see why the unusual order in (3) is needed, because in the above only z^1, z^2 and z^3 are present and so we make z^3 a function of z^4 to bring z^4 into play.

Using this idea, we are able to calculate an expression for $\alpha(-z)_\star \partial_4 \phi - \alpha(-z)_\star \partial_3 \phi$ in terms of the z^i. We carry out this calculation and stop at the second order terms in the z^i. Once that is done we substitute in the obtained expression z^i by its particular values. Lets us remember that we have two cases according to which subset $\mathcal{I}_1$ or $\mathcal{I}_2$ the component of z^i belong (see definition 3).

The two points x_i, x_j are in different sets

Based on this assumption, the component of z^i have to respect the following constraint (where γ and δ are respectively distinct of λ and μ :

$$z^1 = \gamma z^2 + \delta, \quad z^3 = \lambda z^4 + \mu \quad (8)$$

Clearly, the calculation of $\alpha(-z)_\star \partial_4 \phi - \alpha(-z)_\star \partial_3 \phi$ is fairly tedious and we don't explicit it here. Once the substitution of the value of z has been made, there will be terms multiplied by γ, γ^2, $\gamma\delta$, λ etc. The idea is to choose the vector fields a_i and the initial point v_0 to eliminate certain terms and leave only the constant terms and those linear in γ, δ, and so on. To unclutter the equations we use the compact notation:

$$\overline{a_{ij}} = [a_i, a_j] \quad \overline{a_{ijk}} = [a_i, [a_j, a_k]] \quad (9)$$

In this new notation, we omit explicit mention of v_0, but the reader must bear in mind that all of the vector fields are evaluated at the point v_0. As already mentioned, is it a straightforward but long calculation that provides the following. If we calculate the a_i and the initial point v_0 to satisfy the following constraints:

$$\overline{a_{134}} = 0, \quad \overline{a_{114}} - \overline{a_{113}} = 0 \\ \overline{a_{224}} - \overline{a_{223}} = 1, \quad \overline{a_{334}} = 0 \quad (10)$$

where 1 denotes the identity, then the following holds:

$$\begin{aligned}\alpha(-z)_\star(\partial_4-\partial_3)\phi(z) = \quad (11)\\ (a_4-a_3)(v_0)+z^2(\overline{a_{24}}-\overline{a_{23}})+\frac{(z^2)^2}{2}v_0\\ +\gamma(z^2(\overline{a_{14}}-\overline{a_{13}})+(z^2)^2(\overline{a_{124}}-\overline{a_{123}}))\\ +\delta(\overline{a_{14}}-\overline{a_{13}}+z^2(\overline{a_{124}}-\overline{a_{123}}))\\ +\lambda(z^4\overline{a_{34}}+z^2z^4\overline{a_{234}})+\mu(\overline{a_{34}}+z^2\overline{a_{234}})\end{aligned}$$

The two points x_i, x_j are in the same set
The component of z verify the constraint:

$$z^1=\gamma z^2+\delta, \quad z^3=\gamma z^4+\delta \quad (12)$$

and with the particular choice of v_0 expressed by 6:

$$\begin{aligned}\alpha(-z)_\star(\partial_4-\partial_3)\phi(z) = \\ (a_4-a_3)(v_0)+z^2(\overline{a_{24}}-\overline{a_{23}})+\tfrac{(z^2)^2}{2}v_0\\ +\gamma(z^2(\overline{a_{14}}-\overline{a_{13}})+z^4\overline{a_{34}}+\\ (z^2)^2(\overline{a_{124}}-\overline{a_{123}})+z^2z^4\overline{a_{234}}))\\ +\delta(\overline{a_{14}}-\overline{a_{13}}+\overline{a_{34}}+z^2(\overline{a_{124}}-\overline{a_{123}}+\overline{a_{234}})) \quad (13)\end{aligned}$$

Decision test
The interesting thing about (13) and (11) is that if (11) is satisfied then the difference of the last two columns of $\phi_\star$, when pulled back to the origin, is spanned by a 2-dimensional space (once the constant terms are accounted for). However, if (8) is satisfied, then this space is 4-dimensional. Our idea is simply to exploit this difference of dimension in order to determine whether or not two points x_i and x_j belong, or not, to the same data set.

III. Numerical aspects

The implementation of the proposed approach needs discussion about the role played by some parameters: the vector field choice, the dimension of the space connected with the model selection, the measurement noise, the choice of the initial parameters, the allocation of the data to clusters.

A. Optimisation and constraints

For our preliminary trials, we decided to use linear vector fields and so in (2) we have:

$$a_i(v)=A_iv \quad i=1..4 \quad (14)$$

The set of matrices A_i and initial point v_0 were then determined by a standard optimisation routine by transforming equaliies (10) into norms (remember that these are all vectors) and minimising the sum of the norms. The only thing that one needs to be aware of is that constraints need to be added to avoid the routine converging to the zero solution (which of course satisfies (10)). To alleviate this problem we simply added constraints that required $\det A_i \geq \epsilon$ for some $\epsilon > 0$ (concretely we chose $\epsilon = 0.1$). For the considered system, this computation has to be performed once for all. Thus, summarizing, the problem consists in finding A_i and v minimising the following constrained criterion ϕ:

$$\begin{aligned}\Phi = \| \overline{a_{134}}v \| + \| \overline{a_{234}}v \| + \| (\overline{a_{114}}-\overline{a_{223}})v \| + \\ \| (\overline{a_{224}}-\overline{a_{223}})v - v \|\\ \det A_i \geq \epsilon\end{aligned}$$

where $\overline{a_{ijk}}$ are defined in (9) and expressed using the particular field (14).

B. Dimension criteria

We evaluate in (11) and (13) the jacobian matrix of the mapping ϕ using a numerical finite difference method:

$$\partial_i\phi^k(z)\simeq\frac{\phi(z^k+\Delta)-\phi(z^k-\Delta)}{2\Delta}, i=3,4 \quad k=1..4$$

where ϕ is defined in 4 and where z contains a pair of data points (3). We then pull the difference of the two last columns of the jacobian matrix back to the origin using the operator $\alpha(-z)_\star$ (6), again this is done numerically integrating differential equation (2). Then we compare the obtained result with (11) and 13), to see if the space is 2-dimensional or not. In other words if the pulled back difference of the last two columns of $\phi_\star$ is the vector y, the constant part of (11) is the vector b and the vectors multiplied by γ, δ, λ and μ respectively are put into the matrix X. Then $b-y \subseteq \text{span}X$. Obviously, if the relationship $x_i^2=\gamma x_i^1+\delta$ is not satisfied for both data points then this inclusion will not be satisfied.

C. Dealing with noisy data

Of course, the above rank calculation yields only for the case where we have perfect data. In order to increase the interest of the algorithm, we must account for noisy data. The problem becomes one of determining when the matrix $\left[X, b-y\right]$ has rank 1, 2 or 3. In the case that this matrix has rank 1 or 2 then the two data points are in the same set, due to the above mentioned inclusion. If the rank is 3 then the data points are not in the same set. Determining the change of rank of a matrix is notoriously difficult; the experts recommend that to make use of the singular value decomposition (SVD) [7].

D. Initialising the algorithm

As already stated, the data could be (usually are) noisy and so our method should be expected to work within certain reasonable limits. Generally, we begin "blind" with the data set and so an initial data point has to be chosen. Quite obviously, the two regression lines are more than likely to intersect at some point, unless the data are co-linear with only the constants being different. We could expect difficulties in making decisions about membership in one or the other of the two data sets at intersection points. Therefore, to initialise the algorithm to sort the scatter plots, we chose the data point x_q that is in general the furthest from all of the others.

E. Clustering the data points

Once the initial data point x_q is chosen we carry out the above calculations for all the other remaining points. The systems (11) and (13) are compared only for points within a neighbourhood of the initial point. Neighbourhood $\mathcal{N}_{x_k}$ of the point x_k is defined as :

$$\mathcal{N}_{x_k} = \{x_i, \parallel x_i - x_k \parallel \leq \tau_n\}$$

τ_n being a threshold selected by the user. If a neighbouring point looks as though it satisfies (10) then it is included in the same set as the initial point. The algorithm then continues by comparing all the new points in the initial point set with the remaining points and repeats in the same way until there are no more candidate points. At stage (k) in the process, when the cardinality of the initial point set $\hat{S}_1^{(k)}$ is greater than some threshold value (4 in our example), the vector parameters of the local model $p^{(k)} = (\gamma^{(k)}\ \delta^{(k)})$ can be estimated by a standard least squares method [7]. When a point $x_{k+1} = (x_{k+1}^1\ x_{k+1}^2)$ becomes a candidate for inclusion in the initial point set $\hat{S}_1^{(k)}$, then the new values of $p^{(k+1)} = (\gamma^{(k+1)}\ \delta^{(k+1)})$ can be estimated. If the difference between the new and old values $p^{(k+1)}$ and $p^{(k)}$ is too great, then the candidate point is refused, elsewhere the candidate point is accepted. Summarizing, the implementation of the method needs that the user define three parameters: τ_n for the neightbourhood of a given point, τ_s for testing the singular values ratio and τ_p for testing the evolution of the local model parameters.

IV. Example and numerical results

We have tested our method on simulated data and, in this section, we present one example of data sets. For this example we set $x^1 = -0.9 : 0.2 : 0.9$ and $x^2 = 2x^1 + 3 + e$ where $e \sim N(0, 0.04)$ for the first data set and $x^2 = -2x^1 + 2 + e$ where $e \sim N(0, 0.09)$ for the second data set. The x^2 were then normalised to fall within the interval $[-0.9, 0.9]$. In this and the following example we set the algorithm parameters to the following values. For the neighbourhoods test we used $\tau_n = 0.2$, for the singular values test we used $\tau_s = 0.05$ and for the difference test in parameter $p^{(k+1)}$ and $p^{(k)}$ we used $\tau_p = 0.1$ where p_0 is the old vector with γ and δ as components and p_1 the new one.

The results can be seen in Figure 1, the "o" are the points determined as belonging to the first set by the algorithm and the "+" are all the points left when the algorithm has terminated. We can see that, as we expected, the main problems occur at the point where the two curves intersect and it is here that the algorithm starts to follow the other curve but then comes back on course again. Identified parameters for the first model are $2.162, 2.887$ and for the second one $-1.689, 1.946$.

V. Conclusion

In this paper we have presented a feasible solution to the proposed problem of local switching models identification, within certain limits of course. Improvements are to be made to the overall algorithm. The problem is quite a difficult one and we don't believe that one method alone can solve it, rather a combination of methods is required. In this paper we have presented our first attempt at a method, which works reasonably well. More investigation will have to take place in order to develop our ideas further.

We are in the process of analysing our method in order to choose the parameters and thresholds in a more rigorous way. We are also considering a way of choosing the various parameters based on various easily calculated functions of the data, the mean values, variances etc. On a theoritical point of wiev, we are also looking at how to dynamically modify the parameters as the algorithm progresses on a data set in a systematic and optimal fashion.

As our research progresses, we must also address the problem where there are more than two data sets and in the case where the number of sets is not a priori known.

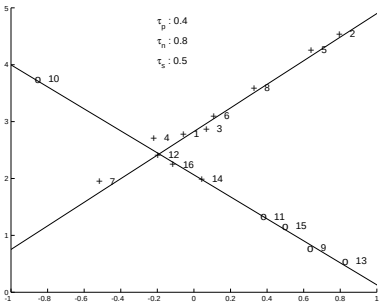

Fig. 1. Results of classification and identification

References

[1] S.-I. Amari and H. Nagaoka (2000), Methods of Information Geometry, Oxford University Press.

[2] M. Basseville and I. Nikiforov (1993), Detection of abrupt changes : theory and application, Prentice Hall Inc.

[3] A.. Bemporad, J. Roll and L. Ljung (2001), Identification of hybrid systems via mixed-integer programming, Proceedings of the 40th IEEE Conference on Decision and Control, p. 786-792.

[4] L. Breiman (1993), Hinging hyperplans for regression, classification and function approximation, Proceedings of the 40th IEEE Conference on Decision and Control, p. 999-1013.

[5] S. Dasgupta, E. Pavlov and Y. Singer (2002), An efficiant PAC algorithm for reconstructing a mixture of lines, Lecture Notes in Computer Science, Springer-Verlag Heidelberg.

[6] A. Isidori (1995), Nonlinear Control Systems, third edition, Springer-Verlag.

[7] G.H. Golub and C.F. Van Loan (1983), Matrix Computations, North Oxford Academic.

[8] M.C. Medeiros, A. Veiga and M.G.C. Resende (1999), A combinatorial approach to piecewise linear time series analysis, Technical report.

[9] R. Murray-Smith and T.A. Johansen (1997), Multiple model approach in modelling and control, Taylor and Francis.

[10] D. Pena and J. Rodriguez (2002), Identifying mixtures of regression equations by the SAR procedure, Bayesian statistic 7, p. 327-348.

[11] R.E. Quandt (1958), The estimation of the parameters of a linear regression system obeying two separate regimes, Journal of the american statistical association, p. 873-880.

[12] B. Strikholm and T. Tersvirta (2002), Determining the number of regimes in threshold autoregressive model using smooth transition autoregression, 13th model selection and evaluation EC2 conference, Bologna.